IN THE SPECIFICATION:

Please amend the three paragraphs on page 1, lines 7 - 27 as shown below.

This application is related to co-pending US Patent Application Serial No.[[_____]] 09/964,999 (IBM Docket No. AUS920010901), entitled APPARATUS AND METHOD OF ASCERTAINING SYSTEMS OPERABILITY BEFORE RUBBING REMOTE COMMANDS herein, filed on even date herewith and assigned to the common assignee of this application.

This application is related to co-pending US Patent Application Serial No.[[_____]] 09/965,001 (IBM Docket No. AUS920010903US1), entitled APPARATUS AND METHOD OF PROVIDING A PLUGGABLE USER INTERFACE by the inventors herein, filed on even date herewith and assigned to the common assignee of this application.

This application is also related to co-pending US Patent Application Serial No.[[_____]] 09/964,998 (IBM Docket No. AUS920010904US1), entitled APPARATUS AND METHOD OF PROVIDING COMMON DISTRIBUTED SERVICES FOR SYSTEM MANAGEMENT APPLICATIONS ACROSS HETEROGENEOUS ENVIRONMENTS by the inventors herein, filed on even date herewith and assigned to the common assignee of this application.

Please replace the paragraph that starts on page 5, line 29 and ends on page 6, line 18 with the following paragraph.

In the depicted example, server 104 is connected to network 102 along with storage unit 106. In addition, clients 108, 110, and 112 are connected to network 102. These clients 108, 110, and 112 may be, for example, personal computers or network computers. In the depicted example, server 104 provides data, such as boot files, operating system images, and applications to clients 108, 110 and 112. Clients 108 ~~109~~, 110, and 112 are clients to server 104. Network data processing system 100 may include additional servers, clients, and other devices not shown. In the depicted example, network data processing system 100 is interconnected via the Internet and represents a collection of networks and gateways that use the TCP/IP suite of protocols to communicate with one another. At the heart of the Internet is a backbone of high-speed data communication lines between major nodes or host computers, consisting of thousands of commercial, government, educational and other computer systems that route data and messages. Of course, network data processing system 100 also may be implemented as a number of different types of networks, such as for example, an intranet, a local area network (LAN), or a wide area network (WAN). Additionally, clients 108, 110 and 112 may be a group or cluster of computers and each cluster may be running under a different operating system (O/S) and having different system management software utilities. Thus, Fig. 1 is

intended as an example, and not as an architectural limitation for the present invention.

Please replace the paragraph that starts on page 6, line 31 and ends on page 7, line 28 with the following paragraph.

With reference now to Fig. 3, a block diagram illustrating a data processing system is depicted in which the present invention may be implemented. Data processing system 300 is an example of a client computer. Data processing system 300 employs a peripheral component interconnect (PCI) local bus architecture. Although the depicted example employs a PCI bus, other bus architectures such as Accelerated Graphics Port (AGP) and Industry Standard Architecture (ISA) may be used. Processor 302 and main memory 304 are connected to PCI local bus 306 through PCI bridge 308. PCI bridge 308 also may include an integrated memory controller and cache memory for processor 302. Additional connections to PCI local bus 306 may be made through direct component interconnection or through add-in boards. In the depicted example, local area network (LAN) adapter 310, SCSI host bus adapter 312, and expansion bus interface 314 are connected to PCI local bus 306 by direct component connection. In contrast, audio adapter 316, graphics adapter 318, and audio/video adapter 319 are connected to PCI local bus 306 by add-in boards inserted into expansion slots. Expansion bus interface 314 provides a connection for a keyboard and mouse adapter 320, modem

322, and additional memory 324. Small computer system interface (SCSI) host bus adapter 312 provides a connection for hard disk drive 326, tape drive 328, and CD-ROM drive 330. Typical PCI local bus implementations will support three or four PCI expansion slots or add-in connectors as shown local PCI expansion slot 332.

Please replace the paragraph that starts on page 15, line 29 and ends on page 16, line 11 with the following paragraph.

Fig. 9 is a dialog window of the groups tab 605 of the common interface. In this window, a user may organize the computer systems in groups. A group is formed by entering a group name in group name box 905 and by adding computer systems to the group using host names box 915 and add button 925. Each computer system added to the group will be shown in host in domain window 920. Any computer system may be taken out of a group by using host names box 915 and remove button 930. Computer systems that have be taken out of the group will be deleted from the list of computer systems shown in host in domain window 920. When a group is complete, it is saved using the save group button 935. Groups can also be formed using the copy group button 940. In this case, two or more existing groups may be combined together. An existing group may be deleted by entering the name of the group in group name box 905 and clicking on delete group box 945. If a group is made of two or more constituent groups, when the name of the group is entered

in group name box 905 the name of the constituent groups will be listed in group members box 950. Browse button 910 is used to list the names of all existing groups.

Please replace the paragraph on page 17, lines 24 - 31 with the following paragraph.

The execution progress window 1000 also contains a "waiting" sub-window 1005, a "working" sub-window ~~1010 and~~ 1010, a ~~"completed" sub-window 1025. The completed sub-window 1025 is further subdivided into~~ "successful" sub-window 1015 and a "failed" sub-window 1020. In the "waiting" sub-window 1005, the names and the number of all the computer systems on which the command has yet to start executing are displayed.

Please replace the paragraph on page 18, lines 10 - 19 with the following paragraph.

When the command has finished executing on a computer system, the name of the computer system will be moved from the "working" sub-window 1010 to ~~the "completed" sub-window 1025 and displayed in~~ either the "successful" sub-window 1015, if it has been successfully completed, or the "failed" sub-window 1020 if it has not successfully completed. The number shown in working window 1010 will be decreased by one and the number in either the "successful'

sub-window 1015 or the "failed" sub-window 1020 will be increased by one.

Please replace the paragraph that starts on page 20, line 22 and ends on page 21, line 7 with the following paragraph.

Returning to Fig. 10, the execution of the command on any computer system may be canceled, if the name of the computer system is highlighted while it is in the "waiting" sub-window 1005 and the stop button 1060 is selected. When this occurs, a window will pop open requesting the user to confirm the cancellation action. If the user does so confirm, the name of the computer system will be taken off the "waiting" sub-window. If the name of the highlighted computer system is instead in the "working" sub-window 1010 ~~1110~~ when the stop button 1060 is asserted, again a window will pop open requesting that the user confirm the cancellation action. If the user does so confirm, the execution of the command will be stopped and the name of the computer system will be moved to the "failed" sub-window. To stop the execution of the command, the software engine sends a stop command to that system. In this case, the reason for the failure may be displayed as "command canceled by user".